AF/IFW

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.: 10/821,431
Filed: April 9, 2004
Inventor(s):
Cypher, et al.

Title: Branch Prediction Scheme Utilizing Multiple Branch Predictors and Multiple Index Hashing Mechanisms for Determining a Final Prediction Based on Intermediaries

Examiner: Vincent Lai
Group/Art Unit: 2181
Atty. Dkt. No: 5181-96100

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date indicated below.

Stephen J. Curran
Printed Name

Signature 3/6/07
Date

REQUEST FOR PRE-APPEAL BRIEF REVIEW

ATTN: BOX AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

Applicant requests review of the final rejection in the above-identified application. No amendments are being filed with this request. This request is being filed with a notice of appeal. The review is requested for the reason(s) stated on the attached sheet(s). Please note that for brevity, only the primary arguments directed mainly to the independent claims are presented, and that additional arguments, e.g., directed to the subject matter of the dependent claims, will be presented if and when the case proceeds to appeal.

Claims 1-4, 6-8, 13-17, 19-20, and 26-27 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Loh (U.S. Patent Publication No. 2005/0223203) (hereinafter "Loh") in view of McFarling (U.S. Patent Publication No. 2001/0056531) (hereinafter "McFarling"). Applicant respectfully traverses this rejection.

Applicant's claim 1 recites

> "A branch prediction mechanism comprising:
>
> a first storage including a first plurality of locations for storing a first set of partial prediction information;
> a second storage including a second plurality of locations for storing a second set of partial prediction information; and

> a control unit configured to perform a first hash function on input branch information to generate a first index for accessing a selected location within said first storage and to perform a second hash function on said input branch information to generate a second index for accessing a selected location within said second storage, wherein said input branch information includes address information corresponding to a fetch address of a current branch instruction;
>
> wherein said control unit is further configured to provide a prediction value based on corresponding partial prediction information in said selected locations of said first and said second storages; and
>
> wherein said control unit is further configured to update said selected locations of said first and said second storages dependent on whether said prediction value yields an accurate branch prediction." (Emphasis added)

The Examiner asserts the combination of Loh and McFarling teach the combination of features recited in Applicant's claim 1. Applicant respectfully disagrees with the Examiner's characterization of both Loh and McFarling, and the allegation that the combination teaches the invention recited in Applicant's claim 1.

More particularly, the Examiner alleges that Loh teaches a first storage including a first plurality of locations for storing a first set of partial prediction information (Loh Fig. 4, element 405 (e.g., branch predictor 1) and a second storage including a second plurality of locations for storing a second set of partial prediction information (Loh Fig. 4, element 405 (e.g., branch predictor 2). Applicant respectfully disagrees. Specifically, Loh teaches at paragraphs [0018], [0019], and [0022]

> "...The prediction history information may be accessed in segments by a number of intermediate branch prediction units 405.
>
> [0019] In one embodiment of the invention, four intermediate branch history units access four segments of branch history from the branch history register. However, in other embodiments, the number of segments and corresponding intermediate branch history units may be greater or fewer than four. In some embodiments of the invention, some intermediate branch history units may be in parallel and others may be in series with any of the parallel branch history units. Furthermore, the series intermediate branch history units may perform intermediate branch predictions in parallel with each other in other embodiments of the invention.
>
> [0022] FIG. 5 is a flow diagram illustrating a method for performing at least one embodiment of the invention. In operation 501, a number of branch prediction segments are accessed in parallel. At operation 505, a number of intermediate branch predictions are performed based off of the branch prediction segments, in which each intermediate branch prediction is based off of a different branch history segment and each branch history segment is smaller than the sum of the branch history segments. At operation 510, a final branch prediction is made based off of the intermediate branch predictions." (Emphasis added)

From the foregoing, Applicant asserts Loh merely teaches the branch predictor units 405 making predictions using information obtained from the segment registers 401, and the final predictor making a prediction based off the intermediate predictions. Applicant submits Loh does not teach disclose or suggest that the branch predictor units 405 have any storage capacity whatsoever. Moreover Applicant cannot find any reference in Loh, whether implied or otherwise, that each predictor 405 may be a storage including a plurality of locations for storing a set of predictions. Applicant again notes that Loh does not provide any specific detail with regard to the structure of elements 405, other than what is discussed above.

In the Advisory action dated March 2, 2007 the Examiner asserts "the Examiner contends the branch predictor units 405 must have some kind of storage capacity. In paragraph 18, Loh teaches that branch history is accessed, meaning there must be some sort of storage element available for the branch predictor." Applicant submits that just because the Examiner contends something is true does not mean that it is without some reasonable teaching in the art. Applicant disagrees with the Examiner and submits Loh actually discloses at paragraph [0018]

> "a prediction history register 401, in which prediction history is stored in one embodiment of the invention. The prediction history register may also be a memory location instead of a register within the processor or some combination thereof." (Emphasis added)

Thus, the branch history information is stored in the prediction history registers 401 and not the branch predictor 405. In addition, as illustrated in FIG. 4, Loh discloses one segment register 401 per branch predictor unit 405.

Thus, Applicant submits Loh **does not teach or suggest** "a first storage including a **first plurality of locations for storing a first set of partial prediction information**," and "**a second storage** including a **second plurality of locations for storing a second set of partial prediction information**," as recited in Applicant's claim 1.

The Examiner acknowledges that Loh does not disclose the hashing functions as recited in Applicant's claim 1. However, the Examiner asserts McFarling teaches the limitation. Applicant respectfully disagrees. More particularly, the Examiner asserts McFarling teaches, at paragraph [0026], lines 10-13; and fig. 14, paragraphs [0087]-[0089], a control unit configured to perform a first hash function on input branch information to generate a first index for accessing a selected location within said first storage, and to perform a second hash function on said input branch information to generate a second index for accessing a selected location within said second storage.

In addition, the Examiner asserts the case of performing multiple has functions in which multiple storages can be accessed is shown. Applicant notes that while McFarling teaches branch predication, and the use of hash functions, Applicant respectfully disagrees with the Examiner's assertion that McFarling teaches the limitations as recited in Applicant's claims. It appears the Examiner is asserting that McFarling is describing the use of a single hash function (e.g., fig. 5, para [0026]), and the multiple hash functions (2) shown in fig. 14 are an extension of that. Applicant disagrees, and notes that McFarling does teach using a hash function in FIG. 5. However, Applicant asserts this is the GShare branch prediction mechanism described in Applicant's background section. Further, Applicant asserts the use of the multiple hash functions as shown in Fig. 14 of McFarling, is not a direct extension of the use of one hash function.

Specifically, McFarling teaches

> "The present invention provides a serial branch predictor that includes a first component predictor operating according to a first algorithm to predict an action, and any number of subsequent component predictors operating according to alternate algorithms to predict the action. ... Each subsequent stage therefore focuses on correction of predictions made by a prior stage. In the preferred embodiment, known as the SerialBLG predictor, the first predictor algorithm is bimodal, a second predictor algorithm is local,

> and a third predictor algorithm is global. Further, each stage is improved according to various methods." (*See* Summary) (Emphasis added)

McFarling also illustrates in FIG. 14, and teaches at paragraphs [0085]-[0091]

> "Global Stage Indexing
>
> [0086] FIG. 14 shows the global stage predictor used in the best mode SerialBLG predictor, which makes use of the stew code, conditional prediction, and partial dominance. The global information is stored in a stew register 140 as described above. The stew register value is XOR'ed with the address of the branch on line 141 to produce V on line 142: V=Stew XOR Addr.
>
> [0087] ...[0088] The value V.sup.+ provides the basis for making the global stage prediction. V.sup.30 could be used to directly access an array of counters, however, as suggested above, most of these counters would be unnecessary. A better approach is to simulate a large array of counters using a cache mechanism 144. On a cache miss, it is assumed that the unavailable count would agree with the prior stage prediction. Unavailable counters are added to the cache only if the final prediction value on line 145 is wrong.
>
> [0089] On such a replacement, the appropriate tag is set to correspond to the V.sup.30 value, and the counter 146 is initialized to weakly agree with the branch causing the miss. If the cache in the global predictor stage uses LRU replacement, the LRU order is only affected when a counter is useful, i.e. generates a better prediction than the earlier stage.
>
> ...
>
> [0091] The method of indexing the global predictor stage cache is particularly significant. For good performance, set-associative caches require addresses that spread out fairly uniformly across the cache. For example, with a 2-way set-associative cache, if everything maps to the same set, then no more than two things can be stored, no matter how large the cache is. The V.sup.+ value, even using the stew code, is still less uniformly distributed than is preferred. To improve performance, two steps can be taken. First, the high order bits on line 147 of V.sup.+ are more random than the low order bits on line 148 because the high order bits on line 147 are a function of a long sequence of branches, whereas the low order bits on line 148 are a function only of the last few branches. It is therefore better to use the low order bits on line 148 for the tag value T on line 149. Second, to further increase the randomness of the remaining bits, the tag value T on line 149 can be XOR'ed into these remaining bits, Z on line 147, to obtain the set index I on line 150:..."

From the foregoing, it appears McFarling is using two hash functions (XOR) to access a single cache structure 144. In addition, Applicant notes the second hash function is operating on at least a portion of the result from the first hash function. This is in contrast to the assertions made by the Examiner, and to Applicant's recited claim language.

In the Advisory action dated March 2, 2007, the Examiner noted "McFarling is used to teach the obviousness of using hashing functions and that it is unreasonable to believe that McFarling and Loh can be combined without adaptations. Loh teaches multiple cache structures that can utilize hash functions and thus to properly combine Loh and McFarling, one having ordinary skill in the art would utilize more than one hash function."

Applicant further disagrees with the Examiner's above and previous assertions. Applicant does not doubt the utility of using a hashing function. However, as disclosed in Applicant's background, there are index aliasing problems associated with the GShare mechanism of McFarling, which Applicant addresses. In addition, as discussed above, McFarling uses the hashing functions differently than Applicant, regardless of whether the Examiner concedes the point, McFarling does, in fact, use the two hash functions to access one cache structure, by hashing different information for a different reason. Thus, Applicant submits the Examiner has not established a *prima facie* case of obviousness simply by showing that McFarling uses hash function(s). Applicant further submits that due to the above-described differences and using hindsight obtained by Applicant's disclosure, the combination of Loh and McFarling could not be used without undue experimentation.

Applicant submits neither Loh nor McFarling, taken either singly or in combination, teach or suggest the combination of features recited in Applicant's claim 1. Accordingly, for the reasons given above, Applicant submits claim 1 patentably distinguishes over Loh and McFarling.

Applicant's claims 14 and 27 recite features that are similar to the features recited in Applicant's claim 1. Thus, for at least the reasons given above, Applicant submits claims 14 and 27 patentably distinguish over Loh and McFarling.

CONCLUSION

Applicant submits the application is in condition for allowance, and an early notice to that effect is requested.

If any fees are due, the Commissioner is authorized to charge said fees to Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C. Deposit Account No. 501505/5181-96100/SJC.

Respectfully submitted,

Stephen J. Curran
Reg. No. 50,664
AGENT FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert, & Goetzel, P.C.
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800

Date: March 6, 2007